POZNAN UNIVERSITY OF TECHNOLOGY

EUROPEAN CREDIT TRANSFER AND ACCUMULATION SYSTEM (ECTS)

COURSE DESCRIPTION CARD - SYLLABUS

Course name
Machine Learning with Graphs [S2SI1E>GUM]

Course

Field of study
Artificial Intelligence

Year/Semester
2/3

Area of study (specialization)
–

Profile of study
general academic

Level of study
second-cycle

Course offered in
English

Form of study
full-time

Requirements
elective

Number of hours

Lecture
15

Laboratory classes
15

Other (e.g. online)
0

Tutorials
0

Projects/seminars
0

Number of credit points

2,00

Coordinators

dr Paweł Misiorek
pawel.misiorek@put.poznan.pl

Lecturers

Prerequisites

A student starting this course should have basic knowledge of mathematics, including the basics of linear algebra, probability, and graph theory. Moreover, she/he should know the fundamental concepts and rules of machine learning, statistics, and data analysis as well as should have programming skills. The student should also be capable of knowledge acquisition from selected sources, as well as express the readiness for collaborating as part of a team.

Course objective

The course is focused on graph-based techniques of mining complex data and their applications to machine learning problems. In particular, the course introduces the concept of hypergraphs as a generalization of graphs in which hyperedges may contain any number of nodes, enabling more accurate modeling of real-world networks and data structures. The course includes the following topics: basics of graph and hypergraph modeling, random graph/hypergraph models and their applications, network mining techniques such as centrality measures or finding most important nodes, community detection algorithms, modularity, graph and hypergraph embeddings, basics of graph neural networks and their applications.

Course-related learning outcomes

Knowledge
K2st_W3: has advanced detailed knowledge regarding selected issues in artificial intelligence and related fields
K2st_W4: has knowledge about development trends and the most important cutting edge achievements in computer science, artificial intelligence and other selected and related scientific disciplines

Skills
K2st_U1: is able to obtain information from literature, databases and other sources (both in Polish and English), integrate them, interpret and critically evaluate them, draw conclusions and formulate and fully justify opinions
K2st_U3: is able to plan and carry out experiments, including computer measurements and simulations, interpret the obtained results and draw conclusions and formulate and verify hypotheses related to complex engineering problems and simple research problems
K2st_U5: can - when formulating and solving engineering tasks - integrate knowledge from different areas of computer science and artificial intelligence (and if necessary also knowledge from other scientific disciplines) and apply a systemic approach, also taking into account non-technical aspects
K2st_U16: can determine the directions of further learning and implement the process of self-education, including other people

Social competences
K2st_K2: understands the importance of using the latest
knowledge in the field of computer science and artificial intelligence in solving research and practical problems
K2st_K4: is aware of the need to develop professional achievements and comply with the rules of professional ethics

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes are verified as follows:
Lectures: The grade based on the assessment test conducted at the last lecture. The test will include several practical tasks and a few theoretical questions concerning the scope of lectures (the list of issues used to state the questions and tasks will be sent to students two weeks before the test). Based on points achieved, a standard scale is used to derive the final mark. The passing threshold: 50%.

Laboratory classes: Students solve practical or programming assignments and report their solutions within two weeks. Each task is evaluated using a scale from 0.0 (lack of the solution) to 5.0 (full solution). The final grade is computed as an average from the individual grades.

Programme content

The course scope covers techniques for modeling complex networks using graphs and hypergraphs, including:
- open libraries for building, processing and visualizing graphs and hypergraphs,
- random models for graphs and hypergraphs,
- techniques for building synthetic networks using graphs,
- methods for describing graph characteristics including measures such as centrality measures, degree correlations, etc.
- community detection and analysis, modularity of graphs and hypergraphs,
- graph embedding techniques,
- basics of graph neural networks.
- examples of practical applications of graphs and hypergraphs in machine learning.

Course topics

The scope of the course covers theoretical (lecture) and practical (laboratory) aspects of the following issues:
1) Basics of graph and hypergraph modeling. Open-source libraries for graph/hypergraph building, processing, and visualization (including iGraph, hypernetx, networkx)
2) Complex network modeling methods. Random graph models. Random hypergraph models.

3) Building synthetic and null models based on random graphs.
4) Network mining techniques such as indicating node centrality, finding the most important nodes, or investigating degree correlations.
5) Community detection algorithms, including the problem of overlapping communities. Graph and hypergraph modularity.
6) Graph embeddings. Measures for embedding quality evaluation.
7) Basics of Graph Neural Networks.
8) Summary of graph applications in machine learning for data cleaning, feature importance, feature design, link prediction, node classification, and graph classification.

Teaching methods

Lectures: multimedia presentations, demonstration of examples with solutions, code analysis

Laboratory classes: practical coding exercises (using Python scripts and open-source libraries), discussion on exemplary solutions, teamwork.

Bibliography

Basic:

Bogumił Kamiński, Paweł Prałat, François Théberge: Mining Complex Networks, Chapman and Hall/CRC, 2021.

William L. Hamilton, Graph Representation Learning, Morgan & Claypool Publishers, 2020. (https://www.cs.mcgill.ca/~wlh/grl_book/)

Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, 2012.

Additional:

Albert-László Barabási, Network Science, Cambridge University Press, 2017.
M. Newman. Networks. Oxford University Press, 2018

A.N. Langville, C.D. Meyer. Google's PageRank and beyond: The science of search engine rankings. Princeton University Press, 2011

F. Chung, L. Lu, Complex Graphs and Networks, American Mathematical Society, 2006.

Breakdown of average student's workload

	Hours	ECTS
Total workload	50	2,00
Classes requiring direct contact with the teacher	30	1,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	20	1,00